

# Monte Carlo Analysis of Local Cross-Correlation ST-TBD Algorithm

Przemyslaw Mazurek<sup>1</sup>[0000-0001-6403-4160] and  
Robert Krupinski<sup>2</sup>[0000-0002-8575-414]

<sup>1</sup> Department of Signal Processing and Multimedia Engineering,  
West Pomeranian University of Technology Szczecin, Szczecin, Poland

[przemyslaw.mazurek@zut.edu.pl](mailto:przemyslaw.mazurek@zut.edu.pl)

<sup>2</sup> [robert.krupinski@zut.edu.pl](mailto:robert.krupinski@zut.edu.pl)

**Abstract.** The Track-Before-Detect (TBD) algorithms allow the estimation of the state of an object, even if the signal is hidden in the background noise. The application of local cross-correlation for the modified Information Update formula improves this estimation for extended objects (tens of cells in the measurement space) compared to the direct application of the Spatio-Temporal TBD (ST-TBD) algorithm. The Monte Carlo test was applied to evaluate algorithms by using a variable standard deviation of the additive Gaussian noise. The proposed solution does not require prior knowledge of the size or measured values of the object. Mean Absolute Error for the proposed algorithm is much lower, close to zero to about 0.8 standard deviation, which is not achieved for the ST-TBD.

**Keywords:** Track-Before-Detect · Tracking · Algorithm Analysis · Monte Carlo · Cross-Correlation.

## 1 Introduction

Tracking algorithms are applied in numerous civil and military applications [6]. Detection algorithms could be used to estimate basic object parameters, such as only position. Tracking allows you to combine subsequent measurements into paths. Tracking allows the filtering of incoming signals to reduce noise and predict the state of the objects. Such filtering is very important because false measurements lead to distortions in tracking. The signal strength associated with several close observations is not an appropriate criterion for choosing a particular observation as an object signal [6].

Different types of measurement data can be processed: radar or video signals are typical. The complexity of tracking systems depends on SNR (Signal-to-Noise Ratio) and the number of tracked objects [7], and real-time tracking of a single object in low SNR scenarios is very sophisticated. It is well known that the Detection and Tracking (classical) approach can only be used in cases of high and medium SNR [6]. Detection based on the threshold algorithm leads to a large number of false detections, so further processing of the tracking part

is not possible. The tracking algorithm (e.g. Benedict–Bordner, Kalman, and EKF) enables suppression of false detection only to a certain level using a motion model [5]. The selection of a fixed or adaptive threshold algorithm is not a solution to the problem. A too low threshold level leads to an increase in the number of observations that are treated as potentially possible. A too high threshold level leads to the omission of observations that may be associated with the object.

An alternative approach is based on Tracking and Detection (Track–Before–Detect). It uses tracking of all possible trajectories without first detecting objects, even if no object is in the range. This is possible due to the processing of raw measurements, without thresholding, as in classical systems [24]. The measured values are accumulated on trajectories, so the signal is filtered (improved SNR), and then detection is possible after tracking. The computational cost of such a solution is extremely high, but in many applications it is acceptable due to safety and expected reliability.

### 1.1 Related Work

There are many TBD algorithms (e.g. Velocity Filters [10], Viterbi TBD [22], ST–TBD [17], SLRT [24], Directional Filters [26]) and all of them support tracking of many objects. This is an internal feature of TBD algorithms. The cost of calculations for most TBD algorithms is constant, regardless of the number of objects and requires high computing power. The reduction of the number of calculations by reducing the number of analyzed possible paths is available in the Particle Filters TBD algorithms [25]. Unfortunately, the main problem with the Particle Filters TBD method is the difficult initialization of the algorithm, especially when the object can appear in range at any time.

A typical tracking system assumes zero mean noise and a positive value for a large point object, so a single pixel (or cell) is excited by the tracked object [24]. Otherwise, additional preprocessing is required for conditioning the input signal [16, 18, 24]. Tracking a larger object that occupies dozens of pixels, sometimes with values below the zero value, requires a dedicated preprocessing algorithm. Large extended objects are considered in Section 2 Objects with known signal characteristics (profile) can be improved by applying matching filters as shown in [16]. Very specific objects are noise objects (only noise is observed from the object), therefore local signal distributions are used [18, 19].

### 1.2 Content and Contribution of the Paper

The proposed solution for tracking extended objects uses the modified ST–TBD with local analysis of cross–correlation between neighboring measurements.

The ST–TBD algorithm is oriented to the processing of individual cells (pixels) of the input signal (image), and the use of cross–correlation allows better use of information about neighbor values in determining the potential position for

the next measurement. This solution can be applied to extended objects without significantly increasing the computing cost and losing information about the similarity of the pixels representing the object in motion.

Using the local cross-correlation and by comparing the current and previous image frame, it is possible to sharpen the measurement values, which is necessary to improve the SNR, as discussed in Section 3. The analysis of the algorithm's performance is possible only through the Monte Carlo approach, as presented in Section 4. The discussion is presented in Section 5. The final conclusions and further work are presented in Section 6.

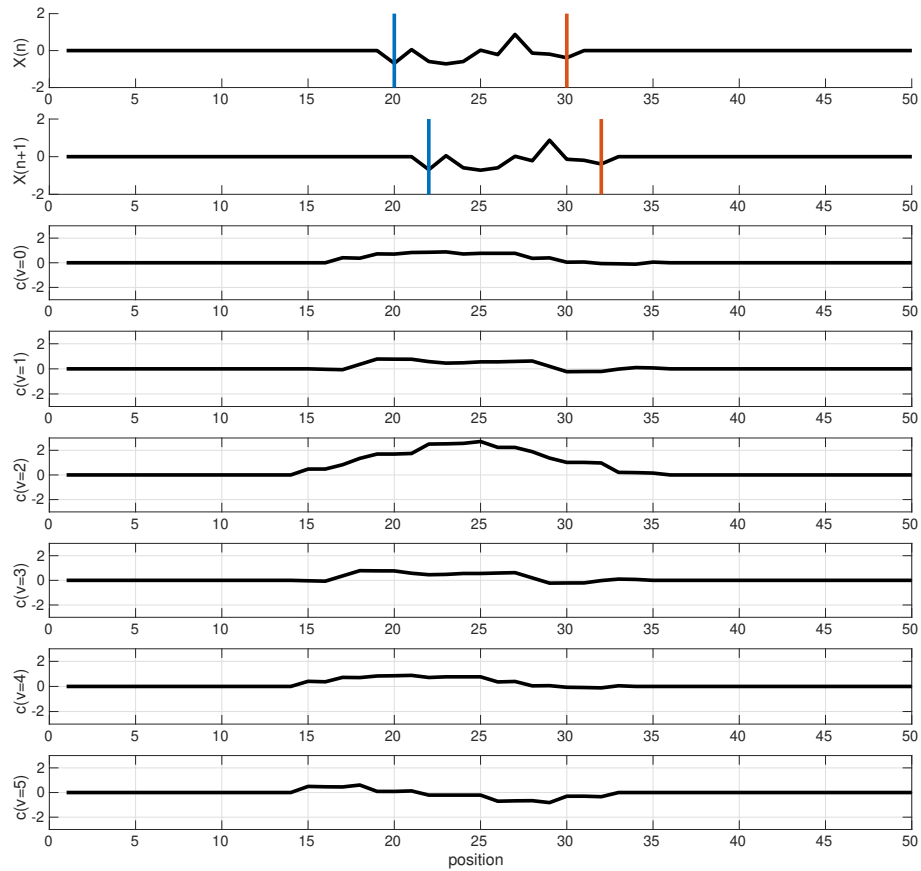
## 2 Data

There are several types of objects being tracked in image processing applications and a typical area of objects with a single pixel. Sometimes the neighborhood pixels are excited because of the imperfections of the sensor. An extended type object includes tens or more pixels, so the direct use of TBD algorithms is not possible, because object features can be treated as separate objects. Classic algorithms use the conversion of extended objects to a single pixel or the estimation and processing of positions. An example of an extended object is shown in Fig. 1 that may have low contrast in poor weather or lighting conditions. The obtained image may also be deteriorated due to the measurement noise caused by poor lighting, long distance and sensor noise.

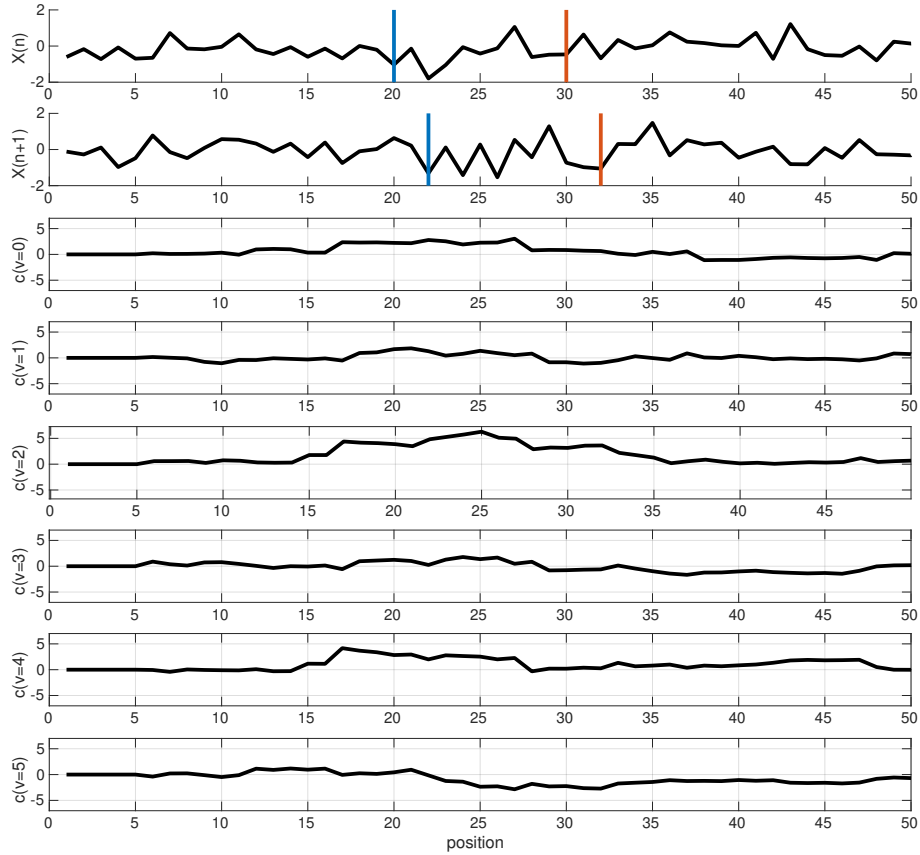


**Fig. 1.** Example of good quality measurement (left), low contrast measurement (center), and measurement of the noisy low contrast (right) for the aircraft.

An additional problem with TBD is blurring caused by the Motion Update formula (explicitly or not). Sharpening is a good solution, but in the case of a low SNR it leads to the emphasis of noise. The solution proposed in this article uses local correlation to sharpen. The example of a cross-correlation that shows the use of local cross-correlation is shown in Fig. 2 and Fig. 3. The highest peak value is for  $c(v = 2)$  in the 25 position, which results in the appropriate spatial offset ( $v = 2$ ) (the object's velocity is 2) and the position of the object. The test case with no background noise is shown in Fig. 2, so detection and tracking is possible using very simple algorithms. Detection and tracking is possible due to the contrast between the object signal and the background. Tracking an object when the signal values for the object are close to the background noise are shown in Fig. 3.



**Fig. 2.** Example of local cross-correlations  $c(\cdot)$  for two 1D  $X(n)$  and  $X(n + 1)$  signals (two time moments:  $n$  and  $n + 1$ ) with an object (between vertical lines) for several possible shifts of objects in space. No background noise.



**Fig. 3.** Example of local cross-correlations  $c(\cdot)$  for two 1D  $X(n)$  and  $X(n + 1)$  signals (two time moments:  $n$  and  $n + 1$ ) with a low contrast object (between vertical lines) for several possible shifts of objects in space.

In this article, the Monte Carlo method [20] was used to compare the basic ST-TBD algorithm with the local cross-correlation ST-TBD for 1D signals. The extension to 2D cases (images) is possible, but is not taken into account due to the high cost of simulation.

It is assumed that the extended object has a size of 11 pixels (cells). The pixel values of the object are randomly selected at the beginning of the test (obtained from a uniform random number generator). The object velocity is an integer value in the range 0 – 10 and the velocity value is chosen randomly. A velocity value of 0 corresponds to the static position of the object (no movement).

The measurement is disturbed by the additive Gaussian noise. The standard deviation of this noise is controlled, configured in a random number generator, which allowed testing of various SNR cases.

There are 1500 cells associated with the position, so the input image has a resolution  $1 \times 1500$  for the assumed 1D case. This allows you to test the object for 100 frames, because 100 frames for a maximum velocity of 10, gives a maximum of 1000 pixels of movement. Estimated and known positions are compared after 100 frames to determine the position error.

### 3 Method

The proposed tracking solution is based on the ST-TBD algorithm that uses the Motion Update formula (2) to predict the state and Information Update formula (3) to process a new input signal:

**Start**

$$P(k = 0, s) = 0 \quad (1)$$

**For**  $k \geq 1$  and  $s \in S$

$$P^-(k, s) = \int_S q_k(s|s_{k-1})P(k-1, s_{k-1})ds_{k-1} \quad (2)$$

$$P(k, s) = \alpha P^-(k, s) + (1 - \alpha)X(k) \quad (3)$$

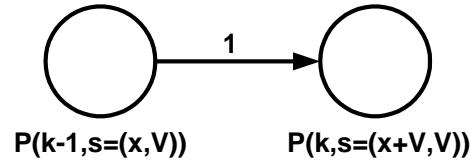
**EndFor**

**End**

where:  $S$  is a state-space,  $s$  is a state (spatial and velocity components),  $k$  is a time step,  $\alpha$  is a smoothing coefficient  $\alpha \in (0, 1)$ ,  $X(k)$  is an observed value,  $P(k, s)$  is the estimated value of state,  $P^-(k, s)$  is the predicted value of state,  $q_k(s|s_{k-1})$  denotes the transition between states (a Markov matrix [24]).

The Markov matrix determines the state changes and can generally describe the probability of changing position and velocity to any other. This matrix can be additionally variable over time. The Markov matrix does not support the transition between velocities in this test, because the assumed velocity of the object is constant. The transition model for a single cell is shown in Fig. 4. The

variable velocity of the object causes that the transition leads to several neighboring targets and therefore the blur of the state space occurs. The knowledge of the object's movement enables optimization of this matrix and reduction of blur.



**Fig. 4.** The simplest transition model.

The local cross-correlation ST-TBD uses a different Information Update formula (3):

$$P(k, s) = \alpha P^-(k, s) + (1 - \alpha) C^N [X(k, s), X(k - 1, s)], \quad (4)$$

where:  $C^N [.,.]$  is a cross-correlation function for a local window with a length of  $N$ .

The state-space can be defined in various ways. In the considered approach, the direct mapping of the input signal to the state-space is selected. One-dimensional measurement uses the following formula:

$$\begin{aligned} C^N [X(k, s = (x, V)), X(k - 1, s = (x, V))] &= \\ &= \sum_{i=-\frac{N-1}{2}}^{\frac{N-1}{2}} X(k, x + i) X(k - 1, x + i - V), \end{aligned} \quad (5)$$

where  $x$  is a position, which leads to a 2D state-space. The local window length is  $N$ . Velocity is the number of cells for the movement of local window between the time moments  $k$  and  $k + 1$  and is denoted by  $V$ . A similar formula can be obtained for 2D inputs (images) and this leads to a 4D state-space.

## 4 Results

The Monte Carlo test can be used to analyze the performance of the algorithm. This method is based on calculating the results for different sets of values (vectors) related to the object signal as well as various random vectors associated with the background signal. A single test cannot be used to evaluate the algorithm, even if the algorithm is deterministic. The problem is the influence of random input data on the output results, so the algorithm evaluation requires many tests for the same statistical parameters, such as standard deviation of the background input noise. The vector of the signal associated with the object is

also generated using a random number generator. The object velocity is another parameter that is randomly selected. The evaluation of the algorithm requires many repetitions and the average results related to the position error could be presented for the comparison of different algorithms or the configurations of a single algorithm. The smooth curves of this error are observed if an acceptable convergence is achieved. The noisy curves are obtained if convergence is not achieved and such results cannot be used to formulate conclusions. The Monte Carlo algorithm is a simple approach and more advanced sampling algorithms are also available, such as MCMC (Markov Chain Monte Carlo) [23].

Standard ST-TBD and local cross-correlations are compared (Fig. 5). In each variant there are 10,000 test repetitions and two smoothing factors are tested: 0.95 or 0.98. The local cross-correlation algorithm is tested with the local window  $N = 11$  and  $N = 21$ .

The fixed velocity and one-dimensional tracking case were adopted with a non-negative motion vector to simplify the calculation. The maximum value of the space-state is assumed as the detection criterion, therefore the location and velocity of the extended object are estimated. The estimation error is a function of additive Gaussian noise, which is the main disruptive factor. The obtained results are shown in Fig. 5. The corrected ST-TBD algorithm assumes detection of any pixel of the extended object instead of the center, which is important due to the random value of the signal object. The correction effect on the result is noticeable as a vertical shift in the result graph, however it is very small (Fig. 5 top-left).

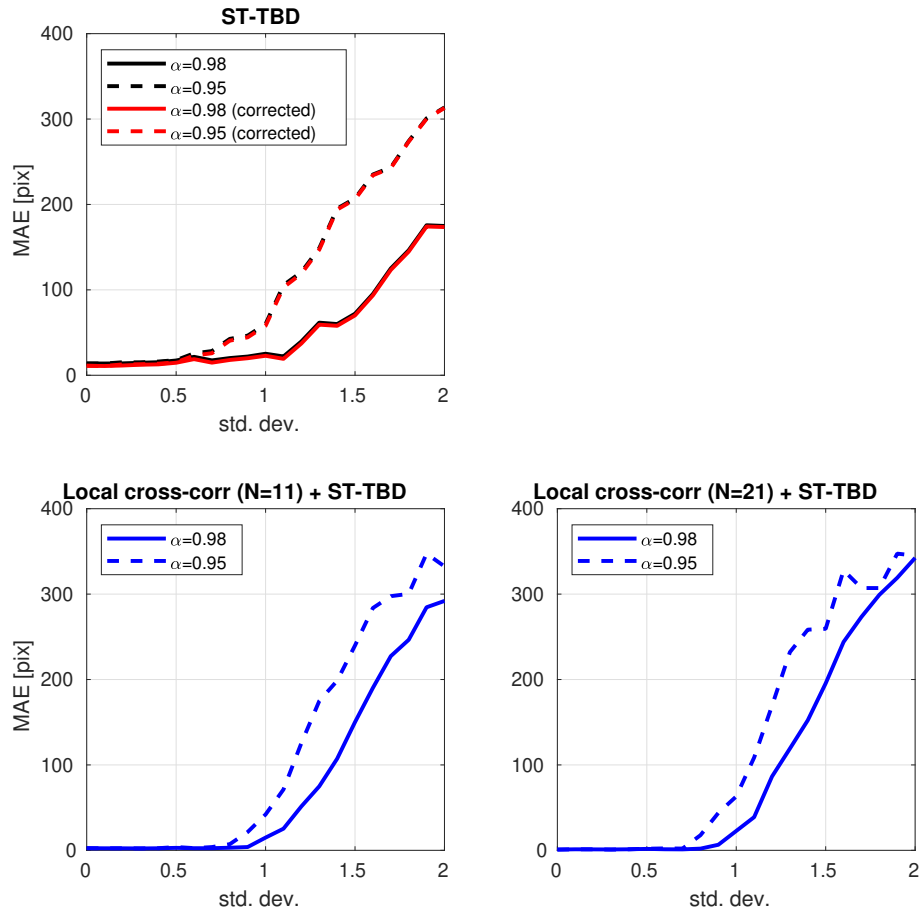
## 5 Discussion

The number of test cases has been selected to obtain smooth curves (Fig. 5). The convergence was analyzed using a variable number of tests, and 10,000 tests were sufficient to determine the properties. The advantage of the Monte Carlo test is the possibility of comparative testing of various algorithms, the influence of parameters and responses for different classes of tracked objects. The obtained results indicate a significant improvement of the proposed method in relation to the standard algorithm (ST-TBD only).

The standard algorithm is single-pixel oriented, and the denoise of measurements is not as effective as the local cross-correlation between neighboring measurement frames. MAE for the proposed algorithm is much lower, close to zero to about 0.8 standard deviation (Fig. 3), which is not achieved for the standard algorithm, even if a correction is applied. The increase in MAE for standard deviation around 1 is expected behavior. This is a region in which a significant influence of the smoothing factor is observed.

A high value of  $\alpha = 0.98$  allows filtering the noise, but in real scenarios this value can not be very high. The smoothing factor reduces the impact of trajectory changes, so it should be estimated for a specific application. Higher errors are observed in the case of large standard deviations ( $> 1.2$ ). Interestingly,





**Fig. 5.** Mean absolute error (MAE) for the Monte Carlo comparison test: the standard algorithm (ST-TBD only) and local cross-correlation with ST-TBD one.

MAE is lower for the standard algorithm, but high MAE values show a general problem for all algorithms.

The influence of the local window size is not relevant to the considered case. This is important, because the size of the object is in many cases an unknown parameter. The cost of calculations is not significant compared to the overall algorithm. The main cost is the calculation of the Motion Update formula (2). The local cross-correlation is calculated for known fixed velocities  $V$ , so it is fast with a cost similar to the 1D FIR filter for the 1D input data.

The real-time implementation is not considered in this document, but efficient parallel or non-parallel implementations are possible if the Markov matrix is not considered as a typical matrix. It is a sparse matrix and numerous implementation optimization techniques are possible. The proposed Information Update formula (4) only requires the calculation of local cross-correlations. It is possible to use parallel processing including SIMD (Single Instruction, Multiple Data) instructions with MAC (Multiply and Accumulate) instructions.

Processing using OpenMP [8], MPI [12] and CUDA [11, 21] is possible for the considered algorithm. Evaluation of algorithms using the Monte Carlo test is very important, because many computers could be used independently for computations.

Tracking systems should have high noise immunity. Parameters of algorithms should be chosen so that the trajectories considered correspond to the real behavior of objects. This means that it is necessary to examine the application of optimization methods [2, 4, 9, 14], thus increasing the credibility of the system and reducing the already large calculation budget.

The article assumes the comparison of the potential location of the object using a local cross-correlation, but the use of other measures may allow a potential improvement in tracking quality [1] as well as clustering [3, 13, 15].

## 6 Conclusions and Further Work

The proposed algorithm can be extended for 2D measurement spaces (such as video or radar images). ST-TBD, like many other TBD algorithms, enables combining data from many of the same or different types of sensors, which is important for improving the quality of tracking. The local cross-correlation assumes the preservation of the signal from the extended object in the neighborhood measurement frames, even if the values associated with the object are unknown.

The achieved result improves the tracking of the hidden signal in the background noise, which is clearly visible in Fig. 3. Detection of the object's position is not possible directly, but the TBD processing used together with the local cross-correlation allows the detection and estimation of position and velocity.

Further work will be related to the application to 2D tracking scenarios and other TBD algorithms.

## Acknowledgment

This work is supported by the UE EFRR ZPORR project Z/2.32/I/1.3.1/267/05 "Szczecin University of Technology – Research and Education Center of Modern Multimedia Technologies" (Poland).

## References

1. Abualigah, L.: Feature Selection and Enhanced Krill Herd Algorithm for Text Document Clustering. *Studies in Computational Intelligence*, Springer (12 2018)
2. Abualigah, L., Khader, A.T., Said Hanandeh, E.: A combination of objective functions and hybrid krill herd algorithm for text document clustering analysis. *Engineering Applications of Artificial Intelligence* **73** (May 2018)
3. Abualigah, L., Khader, A.T., Said Hanandeh, E.: Hybrid clustering analysis using improved krill herd algorithm. *Applied Intelligence* (05 2018)
4. Abualigah, L.M., Khader, A.T.: Unsupervised text feature selection technique based on hybrid particle swarm optimization algorithm with genetic operators for the text clustering. *The Journal of Supercomputing* **73**(11), 4773–4795 (Nov 2017)
5. Blackman, S.: *Multiple-Target Tracking with Radar Applications*. Artech House (1986)
6. Blackman, S., Popoli, R.: *Design and Analysis of Modern Tracking Systems*. Artech House (1999)
7. Boers, Y., Ehlers, F., Koch, W., Luginbuhl, T., Stone, L., Streit, R.: Track before detect algorithms. *EURASIP Journal on Advances in Signal Processing* (2008)
8. Chapman, B., Jost, G., Pas, R.v.d.: *Using OpenMP: Portable Shared Memory Parallel Programming (Scientific and Engineering Computation)*. The MIT Press (2007)
9. Clerc, M.: *From Theory to Practice in Particle Swarm Optimization*, pp. 3–36. Springer Berlin Heidelberg, Berlin, Heidelberg (2011)
10. Dragovic, M.: *Velocity Filtering for Target Detection and Track Initiation*, vol. DSTO-TR-1406. Weapons Systems Division, Systems Sciences Laboratory (2003)
11. Farber, R.: *CUDA Application Design and Development*. Morgan Kaufmann (2011)
12. Karniadakis, G., Kirby, R.: *Parallel Scientific Computing in C++ and MPI*. Cambridge University Press (2003)
13. Kaufman, L., Rousseeuw, P.: *Finding Groups in Data: an introduction to cluster analysis*. Wiley (1990)
14. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: *Proceedings of ICNN'95 - International Conference on Neural Networks*. vol. 4, pp. 1942–1948 vol.4 (Nov 1995)
15. King, R.S.: *Cluster Analysis and Data Mining: An Introduction*. Mercury Learning & Information, USA (2014)
16. Mazurek, P.: Comparison of different measurement spaces for Spatio-Temporal Recurrent Track-Before-Detect algorithm. *Advances in Intelligent and Soft Computing* **102 - Image Processing and Communications Challenges 3**, 157–164 (2011)
17. Mazurek, P.: Parallel distributed downsampled spatio-temporal track-before-detect algorithm. In: *2014 19th International Conference on Methods and Models in Automation and Robotics (MMAR)*. pp. 119–124 (Sep 2014)

18. Mazurek, P.: Preprocessing using maximal autocovariance for spatio-temporal track-before-detect algorithm. *Advances in Intelligent Systems and Computing* **233: Image Processing and Communications Challenges 5**, 45–54 (2014)
19. Mazurek, P.: Noise objects tracking using multiple order statistics and spatio-temporal track-before-detect algorithm. In: Choraś, R.S. (ed.) *Image Processing and Communications Challenges 8*. pp. 112–119. Springer International Publishing, Cham (2017)
20. Metropolis, N.: *The Beginning of the Monte Carlo Method*. Los Alamos Science (1987), <http://library.lanl.gov/la-pubs/00326866.pdf>
21. Sanders, J., Kandrot, E.: *CUDA by Example: An Introduction to General-Purpose GPU Programming*. Addison-Wesley (2010)
22. Scott, T.A., Nilanjan, R.: *Biomedical Image Analysis: Tracking*. Morgan & Claypool (2005)
23. Spall, J.C.: *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control*. Willey (2003)
24. Stone, L., Barlow, C., Corwin, T.: *Bayesian Multiple Target Tracking*. Artech House (1999)
25. Torstensson, J., Trieb, M.: *Particle Filtering for Track Before Detect Applications*. Master’s thesis, Division of Automatic Control, Department of Electrical Engineering, Linköping University (2005)
26. Zhang, T., Li, M., Zuo, Z., Yang, W., Sun, X.: Moving dim point target detection with three-dimensional wide-to-exact search directional filtering. *Pattern Recognition Letters* **28**, 246–253 (2007)